IN THE CLAIMS:

1. (CURRENTLY AMENDED)A method for generating software for an open drive regulator, comprising the steps:

- providing a plurality of object oriented functions, wherein each object oriented function comprises a standardized software interface;

- selecting at least one object oriented function ~~objects~~;

- compiling code for each selected object oriented function separately;

- linking ~~coupling~~ each of the selected object oriented function ~~objects; and~~ to generate~~ing~~ the software for the open drive regulator from the compiled object oriented functions ~~objects~~.

2. (CURRENTLY AMENDED)The method according to claim 1, ~~wherein function objects are produced, or existing function objects in a basic system are modified on a~~further comprising the step of providing customer-specific ~~basis~~object oriented functions having a standardized software interface.

3. (CURRENTLY AMENDED)The method according to claim 1, wherein ~~the function objects are included in the program sequence either statically in the course of overall generation, or dynamically in the course of reloading~~each object oriented function comprises parameters, alarms messages, initialization routines, and test scripts.

4. (CURRENTLY AMENDED)The method according to claim 1, wherein a ~~configuration view is permanently allocated to the function objects using the system infrastructure~~each object oriented function comprises a jump-in point for customer-specific upgrades.

5. (CURRENTLY AMENDED)The method according to claim 1, ~~wherein a diagnosis view is permanently allocated to the function objects using the system infrastructure~~further comprising the step of initializing each selected object oriented function with either a default values or customized values depending on a selection by a user.

6. (CURRENTLY AMENDED)The method according to claim 2~~1~~, wherein ~~the function objects are selected by means of a system object~~the customer specific object oriented function is provided through a communication bus by either an external or internal source.

7. (CURRENTLY AMENDED)The method according to claim 1, wherein the compiled object oriented functions ~~objects~~ are preprocessed and stored as library routines.

8. (CURRENTLY AMENDED)The method according to claim 1, wherein the software has a system configuration which is stored, and only ~~one~~ a change relating to the relevant object oriented function ~~object~~ can be carried out when a object oriented function ~~object~~ is added or removed.

9. (ORIGINAL)The method according to claim 1, wherein data required for software generation are made available via an interface for connection to a data bus or a smart card.

10. (CANCELLED)A computer program in a computer-legible medium, comprising computer-legible program means for performing the method according to claim 1, and wherein the program is run by an open drive regulator having control electronics.

11. (CANCELLED)An open drive regulator having control electronics which contain means for running a software product produced in accordance with the method according to claim 1.

12. (NEW) A system for generating user specific software on the basis of object oriented functions comprising:

- a library comprising a plurality of object oriented functions, wherein each object oriented function comprises a standardized software interface;

- a selection tool for selecting at least one object oriented function ;

- a compiler for compiling code for each selected object oriented function separately;

- a linker for linking each of the selected object oriented function to generate the software for the open drive regulator from the compiled object oriented functions.

13. (NEW) The system according to claim 12, further comprising at least one interface for providing user specific data and/or user specific object oriented functions.

14. (NEW) The system according to claim 12, wherein the system comprises a microcontroller for generating the user specific software wherein the microcontroller is the target microcontroller for running the user specific software.

15. (NEW) The system according to claim 12, further comprising an object memory for storing the library.

16. (NEW) The system according to claim 12, further comprising a parameter memory for storing system specific parameters.

17. (NEW) The system according to claim 12, further comprising a configuration memory for storing configuration data.

18. (NEW) The system according to claim 14, further comprising a program memory for storing the program to be executed by the microcontroller.